

### Remarks

In the Office Action mailed April 13, 2004:

Claims 1-14 were rejected under 35 U.S.C. § 102(b) as being anticipated by Feeney (U.S. Patent No. 6,072,781).

I. Feeney (U.S. Patent No. 6,072,781)

Feeney is directed to a multi-tasking adapter (title). In Feeney, messages are received from a network and dispatched to the network via FIFO buffers 103 (column 5, lines 65-67). For receive traffic, messages go from receive FIFO buffers 105 to buckets 116 in system memory 114 if buckets are enabled (column 14, line 1-12).

Different FIFO buffers are not employed to re-order messages, but rather to separate messages relating to different media, different media tasks, different priority levels, etc. (column 3, line 64 to column 4, line 2).

**A. Feeney Does Not Store Tags in FIFO Queues**

In claimed embodiments of the present invention (e.g., claims 1, 8, 12), messages from a network are temporarily stored in slots of a message store. For each message, a tag identifying the slot is queued in a FIFO queue. When a tag reaches the head of a FIFO queue, the tag is removed and the message corresponding to the tag is dequeued from the message store.

As described above, Feeney receives incoming messages in a received FIFO buffer 105, from which they may be transferred into system memory 114. Feeney does not queue tags in separate FIFO queues to determine the order in which to process the incoming messages.

The Examiner has compared the system memory of Feeney to Applicant's message store, and Feeney's FIFO buffers to Applicant's FIFO queues. These comparisons fail because the two structures are used in parallel (and to store different things) in the present invention, but are used serially in Feeney (to store the same thing).

Claims of the application have been amended to make these differences clearer.

**B. Feeney Cannot and Does Not Dequeue Messages when Corresponding Tags are at the Head of FIFO Queues**

In claimed embodiments of the present invention (e.g., claims 1, 8, 12), messages from a network are temporarily stored in slots of a message store. For each message, a tag identifying the slot is queued in a FIFO queue. When a tag reaches the head of a FIFO queue, the tag is removed and the message corresponding to the tag is dequeued from the message store.

As described in Section I.A, Feeney does not store tags in FIFO queues. Rather, Feeney stores messages first in receive FIFO buffers, and then in system memory buckets. Therefore, Feeney cannot dequeue messages from a message store based on corresponding tags reaching the head of FIFO queues, for two reasons: (1) Feeney does not store tags in FIFO queues or buffers; and (2) messages are removed from FIFO queues *before* they are ever put into a message store (system memory).

**II. Selected Claims**

**A. Claims 1-7**

Claim 1 has been amended to make it clearer that, in the recited embodiment of the invention, a tag corresponding to a message stored in a slot of a message store is queued in a FIFO queue when the message is stored. The message store and FIFO queues are thus used to store different things at the same time.

As described above in Section I.A, a message in Feeney is first stored in a FIFO buffer, and *then* transferred to a bucket in system memory. Tags are not stored in Feeney, and the FIFO buffers and system memory buckets are used to store the same thing at different times.

Claim 2 recites the dequeuing of a message from a message slot after its corresponding tag is at the head of a FIFO queue. As described in Section I.B, Feeney cannot and does not teach or suggest such action.

**B. Claims 8-11**

Claim 8 has been amended to make it clearer that the FIFO queues are used to store tags *while* corresponding messages are stored in the message store. For example, in the embodiment depicted in FIG. 3 of the present application, a tag is queued in a FIFO queue only after its corresponding message is stored in a slot of the message store.

In Feeney, as described in Section I.A, messages are first stored in a FIFO buffer, and then stored in system memory; the claimed embodiment of the invention employs a FIFO queue and a message store at the same time or even in the reverse order. Thus, Feeney cannot anticipate the embodiment of the invention recited in claim 8.

**C. Claims 12-14**

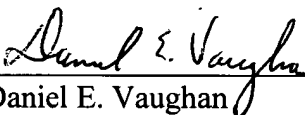
Claim 12 has been amended to make it clearer that, in the recited embodiment of the invention, a tag corresponding to a message stored in a slot of a message store is queued in a FIFO queue only *after* the message is stored. As described above in Section I.A, a message in Feeney is first stored in a FIFO buffer, and *then* transferred to a bucket in system memory. Tags are not stored in Feeney.

**CONCLUSION**

No new matter has been added with the preceding amendments. It is submitted that the application is in suitable condition for allowance. Such action is respectfully requested. If prosecution of this application may be facilitated through a telephone interview, the Examiner is invited to contact Applicant's attorney identified below.

Respectfully submitted,

Date: June 15, 2004

By:  42,199  
Daniel E. Vaughan (Registration No.)

**Park, Vaughan & Fleming LLP**  
702 Marshall Street, Suite 310  
Redwood City, CA 94063  
(650) 474-1973: voice  
(650) 474-1976: facsimile